CiteSeer Find: utilizing Documents Citations

Searching for PHRASE utilizing register pressure make inlining decisions.

Restrict to: Header Title Order by: Citations Hubs Usage Date Try: Amazon B&N Google (RI) Google (Web) CSB DBLP

No documents match Boolean query. Trying non-Boolean relevance query.
1000 documents found. **Only retrieving 125 documents (System busy - maximum reduced).** Retrieving documents... **Order: relevance to query.**

Global Register Allocation Based on Graph Fusion - Guei-Yuan Lueh (1996) (Correct) (7 citations)
Global **Register** Allocation Based on Graph Fusion Guei-Yuan Lueh
www.cs.cmu.edu/afs/cs.cmu.edu/project/iwarp/archive/fx-papers/lcpc96.ps

Flow-directed Inlining - Jagannathan, Wright (1996) (Correct) (36 citations)
cost of passing arguments, saving and restoring **registers**, building return linkage information, and
was masked by effects like increased **register pressure**, and pessimistic assumptions made by the
can be significant, especially for programs that **make** liberal use of data and control abstractions.
ftp.nj.nec.com/pub/pls/pldi96.ps

Towards Better Inlining Decisions Using Inlining Trials - Dean, Chambers (1994) (Correct) (16 citations)
only if the number of instructions in its RTL (**register** transfer language) representation is less than
benefits of **inlining**. Previous automatic **decision make**rs used simple techniques for estimating costs
Orlando, FL, June, 1994. Towards Better **Inlining Decisions** Using **Inlining** Trials Jeffrey Dean
ftp.cs.washington.edu/homes/chambers/auto-inlining.ps.Z

Training Compilers for Better Inlining Decisions - Dean, Chambers (1993) (Correct) (1 citation)
the **inlining** phase of the compiler runs prior to **register** allocation and instruction scheduling, and
Report 93-05-05 May 1993 1 Training Compilers to **Make** Better **Inlining Decisions** Jeffrey Dean and Craig
Training Compilers for Better **Inlining Decisions** Jeffrey Dean and Craig Chambers
ftp.cs.washington.edu/tr/1993/05/UW-CSE-93-05-05.PS.Z

Fast and Effective Procedure Inlining - Oscar Waddell (1997) (Correct) (5 citations)
Fast and Effective Procedure **Inlining** Oscar Waddell and R. Kent Dybvig Indiana
www.cs.indiana.edu/~owaddell/papers/sas-97.ps.gz

A Comparative Study of Static and Profile-Based.. - Arnold, Fink, Sarkar, .. (2000) (Correct) (1 citation)
of I-cache misses, and increase the number of **register** spills with current **register** allocation
decrease in code locality, increase in **register pressure**, etc. Again, for the sake of simplicity, we
they do not use runtime information and thus might **make** poor **inlining decisions**. Conversely, profile-based
www.research.ibm.com/people/p/pfs/papers/dynamo00.ps

Static Interprocedural Optimizations in Java - Zoran Budimlic (1998) (Correct) (2 citations)
tion-any variables whose protection was relaxed **make inlining** possible might compro- mised code added
tion. This paper presents approaches cloning **inlining** that profitably used even single class. These
www.cs.rice.edu/~zoran/CRPCTR98746.ps.gz

Software-Directed Register Deallocation for.. - Lo, Parekh, Eggers, .. (Correct) (2 citations)
software techniques that increase **register** file **utiliz**ation for simultaneous multithreading (SMT)
and Distributed Systems Software-Directed **Register** Deallocation for Simultaneous Multithreaded
www-cse.ucsd.edu/users/tullsen/TPDS99.ps

Inline expansion: when and how? - Serrano (1997) (Correct)
show in [6] that **inlining** for C may increase **register** save and restore operations because of C
permit call sites to become 4 times larger and we **make** each recursive call to the **inlining** algorithm
Inline expansion: when and how? Manuel Serrano
kaolin.unice.fr/~serrano/diffusion/ps/plilp97.ps.gz

Minimum Register Requirements for a Modulo Schedule - Alexandre Eichenberger (1994) (Correct) (4 citations)
overlap, some functional unit can be fully **utilized**, resulting in a schedule with maximum
Minimum **Register** Requirements for a Modulo Schedule Alexandre E.
a scheduling algorithm that reduces the **register pressure** while scheduling for high throughput is
www.cs.rutgers.edu/~uli/cs671/MICRO97-Eichenberger.ps

Optimum Modulo Schedules for Minimum Register Requirements - Eichenberger, Davidson.. (1995) (Correct) (12 citations)
to obtain higher performance by better **utilizing** wider issue machines and reducing the schedule
Optimum Modulo Schedules for Minimum **Register** Requirements Alexandre E. Eichenberger and
www4.ncsu.edu/~alexe/papers/ICS95.ps

Whole-Program Optimization of Object-Oriented Languages - Dean (1996) (Correct) (18 citations)
Knoblock &Ruf 96]Some PIC implementations also **utilize** a move-to-front optimization on a cache hit, in
address, usually by jumping indirectly through a **register**. This is illustrated in Figure2.7. Dispatch
be **inlined** in order to be analyzed increases the **pressure** to **inline** to obtain more class information
ftp.cs.washington.edu/tr/1996/11/UW-CSE-96-11-05.PS.gz

Motivation - Subprogram Inlining (Correct)
Davidson and Holler show that inattention to **register** allocation considerations can cause **inlining** to
frequently executed paths, we would prefer to not **make** the less frequently executed paths exorbitantly
Partial **Inlining** Robert Muth Saumya Debray Department of
www.cs.arizona.edu/people/debray/papers/partial-inlining.ps

Protocol Latency: MIPS and Reality - David Mosberger (1995) (Correct) (2 citations)
from interfering with the d-cache and reducing CPU **utiliz**ation are nevertheless desirable goals. Our
to give outlined code low-priority during **register** allocation. Our present implementation does not
the above list applies)For example, the x-kernel **makes** extensive use of a hash-table manager [HMPT89,
ftp.cs.arizona.edu/reports/1995/TR95-02.ps

Method Inlining in the Titanium Compiler - Cs Semester Project (Correct)
be limited or even detrimental due to increased **register pressure** in the caller and the common inability
or even detrimental due to increased **register pressure** in the caller and the common inability for
of the opaque method call with equivalent code can **make** it easier for intraprocedural optimizations (such
www.cs.berkeley.edu/Research/Projects/titanium/papers/bonachea-method-inlining.pdf

Lambda-Splitting: A Higher-Order Approach to Cross-Module.. - Blume, Appel (1997) (Correct) (17 citations)
representing entire compilation units can also be **utilized** when splitting functions too large to be placed
improves performance by 4-8% on existing code, and **makes** it possible to use much more data abstraction by
Abstract We describe an algorithm for automatic **inline** expansion across module boundaries that works in
www.cs.princeton.edu/fac/appel/papers/inlining.ps

Register Pressure Sensitive Redundancy Elimination - Gupta, Bodik (1999) (Correct)
Register Pressure Sensitive Redundancy Elimination
www.cs.pitt.edu/~gupta/research/Comp/CC99.ps

First 20 documents Next 20

Try your query at: Amazon Barnes & Noble Google (RI) Google (Web) CSB DBLP

CiteSeer Find: savin site Documents Citations

Searching for PHRASE savin site register pressure first compilation inlining decision second compilation.

Restrict to: Header Title Order by: Citations Hubs Usage Date Try: Amazon B&N Google (RI) Google (Web) CSB DBLP

No documents match Boolean query. Trying non-Boolean relevance query.
1000 documents found. **Only retrieving 250 documents (System busy - maximum reduced).** Retrieving documents... **Order: relevance to query.**

Global Register Allocation Based on Graph Fusion - Guei-Yuan Lueh (1996) (Correct) (7 citations)
function. The priority function captures the **saving**s in memory accesses from assigning a **register** to instead of the higher caller-save cost at all call **sites**. When applied to the alvinn program, this Global **Register** Allocation Based on Graph Fusion Guei-Yuan Lueh
www.cs.cmu.edu/afs/cs.cmu.edu/project/iwarp/archive/fx-papers/lcpc96.ps

A Fresh Look to Inlining Decision - Manuel Serrano (1995) (Correct) (1 citation)
this, a new rule has been presented: Rule 5 (**saving** estimations) If the body of f ,after **inlining**, algorithm can choose to **inline** a function on a call **site**, but also to not do it on another call **site**. User Global optimizations (such as inter-procedural **register** allocations [4] or control flow analysis [12]
kaolin.unice.fr/~serrano/diffusion/ps/ics95.ps.gz

Inline expansion: when and how? - Serrano (1997) (Correct)
be **inlined** without increasing the program size. **Saving**s estimates The **inlining** of a call to a function algorithm can choose to **inline** a function at a call **site**, but also not to do it at another call **site**. User show in [6] that **inlining** for C may increase **register** save and restore operations because of C
kaolin.unice.fr/~serrano/diffusion/ps/plilp97.ps.gz

Global Instruction Scheduling In Machine SUIF - Gang Chen (1997) (Correct) (2 citations)
For example, we have a single piece of code for **register** allocation 1 that sees only the salient perform code motions without increasing **register pressure**)Global instruction scheduling consists of information, like the actual specifier of the **first** general-purpose temporary **register** in a
www.eecs.harvard.edu/machsuif/papers/hpca3.ps

Training Compilers for Better Inlining Decisions - Dean, Chambers (1993) (Correct) (1 citation)
its compiled code space cost and estimate its **saving**s in execution time as a result of **inlining**. If **inlining decisions** of the same routine at call **sites** that have similar static information. Central to the **inlining** phase of the compiler runs prior to **register** allocation and instruction scheduling, and
ftp.cs.washington.edu/tr/1993/05/UW-CSE-93-05-05.PS.Z

Deferred Compilation: The Automation of Run-Time Code Generation - Leone, Lee (1993) (Correct) (10 citations)
their arguments. It remains to be seen whether this **saving**s will in general justify the increased **register** function rather than **inlining** its body at each call **site**. This strategy is commonly called specialization many standard **compilation** techniques, such as **register** allocation and **inlining**, can be employed. ffl
www.cs.cmu.edu/afs/cs.cmu.edu/user/mleone/papers/deferred-compilation.ps

Interprocedural Transformations for Parallel Code Generation - Hall, Kennedy, McKinley (1991) (Correct) (17 citations)
If the source or sink of a dependence is a call **site**, a secPage Augmented Call Graph RSD Analysis system consists of two principal phases. The **first** takes place prior to **compilation**. At the end of is provided by an existing interprocedural **compilation** system. We demonstrate the potential of these
softlib.rice.edu/pub/CRPC-TRs/reports/CRPC-TR91149-S.ps.gz

A Data-Stream Language for Protocols - Clayton, Calvert (Correct)
it maps data streams into the host system's **register** set and it replicates the code to fill the

a data-stream architecture for protocol software. **First**, it seems easy to map protocol data processing it with a combination of structuring and **compilation** techniques. The structuring techniques are
www.cc.gatech.edu/computing/Telecomm/playground/dsa/cwc.ps

Compiler-Controlled Multithreading for Lenient Parallel.. - Schauser, Culler, von.. (1991) (Correct) (34 citations)
threads closely together in time and to use **registers** across threads. Remote communication is via returns a pair containing the square of its **first** argument and the product of its two arguments. It multithreaded execution can be addressed as a **compilation** problem, to achieve switching rates
www.cs.cornell.edu/tve/papers-ucb/fpca91.ps.gz

Compilation Techniques for Low Energy: An Overview - Tiwari, Malik, Wolfe (1994) (Correct) (24 citations)
by and large not recognized the potential energy **saving**s achievable through optimization of software high energy costs compared to instructions with **register** operands. Instructions using only **register** energy, some points should be noted in this regard. **First**, compilers do not always choose the most
ftp.ee.princeton.edu/pub/vivek/slpe94.ps

Incremental Recompilation for Standard ML of New Jersey - Harper, Lee, Pfennig, Rollins (1994) (Correct) (9 citations)
generate caches this intermediate output by **saving** it into a file. Note how compileFun calls have been implemented and used daily at a number of **sites**. The current implementation is richer in some by the signature above, but can be extended by **register**ing new types of sources and new tools. 7
www.cs.cmu.edu/afs/cs.cmu.edu/project/fox/mosaic/papers/incremental-recomp.ps

A Survey on Knowledge Compilation - Cadoli, Donini (1998) (Correct) (15 citations)
compilation of propositional knowledge bases. We **first** define and limit the scope of such a technique, 1 A Survey on Knowledge **Compilation** Marco Cadoli and Francesco M. Donini
www.dis.uniroma1.it/pub/Al/papers/cado-doni-97.ps.gz

Interprocedural Array Data-Flow Analysis for Cache Coherence - Lynn Choi (1995) (Correct) (1 citation)
by summarizing the access information at each call **site**. The **second** top-down context analysis allows the 6, 7, 13]The epoch number is stored in an n-bit **register** in each processor, called epoch counter (R cache locality across procedure boundaries. The **first** bottom-up side effect analysis eliminates the
polaris.cs.uiuc.edu/reports/1427.ps.gz

Making Pure Object-Oriented Languages Practical - Chambers, Ungar (1991) (Correct) (61 citations)
times. Also the compiler is biased in favor of **saving** compile time rather than generating better code as SELF Release 1.1, since January 1991. Over 150 **sites** around the world have a copy of our SELF aCol: vector call compiler call compiler 5 **register** allocation. In most traditional allocators
self.sunlabs.com/papers/oopsla91.ps.Z

Generating Efficient Code for Lazy Functional Languages - Smetsers, Nöcker, van.. (1991) (Correct) (20 citations)
normal form the code becomes much faster, for the **saving** and restoring of the **registers** is not needed to be applicable. Some of them (for example **register** allocation algorithms) are common for the
ftp.cs.kun.nl/pub/CSI/SoftwEng.FunctLang/papers/smes91-codegeneration.ps.gz

Optimum Modulo Schedules for Minimum Register Requirements - Eichenberger, Davidson.. (1995) (Correct) (12 citations)
Optimum Modulo Schedules for Minimum **Register** Requirements Alexandre E. Eichenberger and
www4.ncsu.edu/~alexe/papers/ICS95.ps

Optimizing Dynamically-Typed Object-Oriented Languages.. - Hölzle, Chambers, Ungar (1991) (Correct) (7 citations)
through these optimizations often overshadow the **saving**s from just removing the call/return overhead and include more than one cached lookup result per call **site**. For a set of typical object-oriented SELF subexpression elimination, and better global **register** allocation. The benefits obtained through these
self.sunlabs.com/papers/ecoop91.ps.Z

Hardware Compilation of the ProCoS Gas Burner Case Study using.. - Bowen (1996) (Correct)
on the Transputer)However, towards the end of the **first** phase of the project, a new research group at the Hardware **Compilation** of the ProCoS Gas Burner Case Study using

the fact that the **first** component in the **second** circuit being compiled contains the final finish
ftp.cs.reading.ac.uk/pub/formal/jpb/hsvw96.ps

TIL: A Type-Directed Optimizing Compiler for ML - Tarditi, Morrisett, Cheng (1995) (Correct) (116 citations)
associate these tables with the addresses of call **sites** within functions at compile time. When garbage calling conventions (multiple arguments passed in **registers**) and tag-free polymorphic, structural intensional polymorphism comes with two costs. **First**, we must construct and pass representations of
www.cs.cmu.edu/~rwh/papers/til/tr.ps

A Fragment Calculus - towards a model of Separate.. - Drossopoulou.. (1998) (Correct) (9 citations)
Student and CStudent, i.e. B =B st B cs **First**, the class Lab is compiled, i.e. S 1 =S lab
A Fragment Calculus towards a model of Separate **Compilation**, Linking and Binary Compatibility Sophia
CStudent.class and Lab.class. In the **second** phase we add the eld grade of type char to
outoften.doc.ic.ac.uk/projects/slurp/pubs/fragcalc.ps.gz

First 20 documents Next 20

Try your query at: Amazon Barnes & Noble Google (RI) Google (Web) CSB DBLP

CiteSeer Find: savin site Documents Citations

Searching for PHRASE savin site register pressure first compilation inlining decision second compilation.

Restrict to: Header Title Order by: Citations Hubs Usage Date Try: Amazon B&N Google (RI) Google (Web) CSB DBLP

No documents match Boolean query. Trying non-Boolean relevance query.
1000 documents found. **Only retrieving 125 documents (System busy - maximum reduced).** Retrieving documents... **Order: relevance to query.**

An Empirical Evaluation of Knowledge Compilation - Kautz, Selman (1994) (Correct) (1 citation)
In fact, we show substantial total computational **saving**s in using the bounds together with the original not just "skimming off" the easy queries)We will **first** show, by using a general complexity-theoretic An Empirical Evaluation of Knowledge **Compilation** Henry Kautz and Bart Selman AI Principles
akpublic.research.att.com/~kautz/papers-ftp/aaai94kc.ps

Correctness of compiling Occam to Transputer code - Börger, Durdanovic (Correct)
optimization of OCCAM rel\Gammaenv ,obtained by **saving** one son in the **compilation** of PAR statements. the Transputer ground model consisting of various **registers** used for the execution of Transputer For the control structure **compilation** we **first** linearize the flowchart (by introducing goto-
ftp.eecs.umich.edu/groups/gasm/occam2.ps.gz

Dynamo: A Staged Compiler Architecture for Dynamic Program.. - Leone, Dybvig (1997) (Correct) (5 citations)
controlled by an optimization directive at its call **site**, which has access to the contextual information event dispatcher whenever a new event handler is **registered** [6]Our prior research has included an occurs only once per program, when the code is **first** available, dynamic optimization may occur both as
www.cs.indiana.edu/l/www/proglang/dynamo/tr490.ps

Minimum Register Requirements for a Modulo Schedule - Alexandre Eichenberger (1994) (Correct) (4 citations)
Minimum **Register** Requirements for a Modulo Schedule Alexandre E.
a scheduling algorithm that reduces the **register pressure** while scheduling for high throughput is schedule for machines with finite resources. The **first** lineartime algorithm can handle loops whose
www.cs.rutgers.edu/~uli/cs671/MICRO97-Eichenberger.ps

Increasing Memory Bandwidth with Wide Buses: Compiler, .. - Lopez, Valero, Llosa, .. (1996) (Correct)
trade-offs of wide buses and the additional **register pressure**, showing that it is minimal and has of wide buses and the additional **register pressure**, showing that it is minimal and has negligible improve the bandwidth between the processor and the **first**-level cache. Keywords: VLIW and superscalar
ftp.ac.upc.es/pub/reports/DAC/1996/UPC-DAC-1996-57.ps.Z

Overview of the Cecil/Vortex Project - Craig Chambers (Correct)
argument types that occurred at the program's call **sites** this information can then be exploited when has been constructed. As of Spring 1995, a **first** version of the Cecil language has been defined ways of incorporating the profile data into the **compilation** process, including recompiling a call **site**
www.cs.washington.edu/research/projects/cecil/www/www/www/Overview/overview.ps

Fast and Effective Procedure Inlining - Oscar Waddell (1997) (Correct) (5 citations)
guide **inlining decisions** with an estimate of code **saving**s and performance gain that is based on Information about the arguments at a call **site** is used to decide whether **inlining** that call is to the speed and effectiveness of our algorithm: **First**, it is polyvariant. Information about the
www.cs.indiana.edu/~owaddell/papers/sas-97.ps.gz

Stack-Based Typed Assembly Language - Morrisett, Crary, Walker, Glew (1998) (Correct) (24 citations)
Figure 8: Function **Compilation** is not an argument)**saving** the return address, and creating a frame in \Gamma and I ,and are instantiated at the call **site** to the function. As usual, we consider such as tail call elimination and callee-saves **registers**. This paper also formalizes the typing

reports-archive.adm.cs.cmu.edu/anon/1998/CMU-CS-98-178.ps

Iterative Compilation in a Non-Linear Optimisation Space - Bodin, Kisuki.. (1998) (Correct) (4 citations)
not implicitly or explicit modeled, such as the **register** allocation policy of the local compiler, then
information, such as code size, **register pressure**, slot utilisation etc.it is possible to
paper examines another approach to this problem by **first** describing the problem as that of searching a
www-cse.ucsd.edu/users/calder/pfdc/papers/pfdc-bodin.ps.Z

Software-Directed Register Deallocation for.. - Lo, Parekh, Eggers, .. (Correct) (2 citations)
and Distributed Systems Software-Directed **Register** Deallocation for Simultaneous Multithreaded
www-cse.ucsd.edu/users/tullsen/TPDS99.ps

Implementation of Stack-Based Languages on Register Machines - Ertl (1996) (Correct) (1 citation)
set. Otherwise most of the time would be spent **saving** and restoring **registers** at procedure entries
this time with as many arguments as there are call **sites**. Later they will be converted into moves at the
Implementation of Stack-Based Languages on **Register** Machines ausgefuhrt zum Zwecke der Erlangung des
mips.complang.tuwien.ac.at/projects/../papers/ertl96diss.ps.gz

Executable Assertions and Separate Compilation - John Gough (Correct)
a circumstance in which we have thought the modest **saving** in execution time was worthwhile. Our own code
to move tests from called procedure to the call **site**. The gain arises, because the caller of a
?Joint Modular Languages Conference, Linz 2 **First**ly, failure of the assertion at runtime can
www-fs.informatik.uni-tuebingen.de/~klaeren/jmlc97.ps.gz

Register Pressure Sensitive Redundancy Elimination - Gupta, Bodik (1999) (Correct)
of the same value by computing the value once, **saving** it in a temporary, and reusing the value from
Register Pressure Sensitive Redundancy Elimination
www.cs.pitt.edu/~gupta/research/Comp/CC99.ps

Generalizations of the Sethi-Ullman algorithm for register.. - Appel, Supowit (1987) (Correct) (6 citations)
the rest of the subtrees must be evaluated while **saving** the result of subtree p(1) in b p (1)
Generalizations of the Sethi-Ullman algorithm for **register** allocation Andrew W. Appel Kenneth J. Supowit
each node, which of its two children to evaluate **first**. Suppose the optimally ordered computation of the
www.cs.princeton.edu/~appel/papers/sun.ps

Loop Scheduling Algorithm for Timing and Memory Operation.. - Fei Chen (Correct)
1)Consequently, we need to reserve **registers** for **saving** every copy of node 3 for many iterations in a
for Timing and Memory Operation Minimization with **Register** Constraint Fei Chen Sissades Tongsima Edwin
www.nd.edu/~esha/papers/fei/SiPS98.ps

Incl: A Tool to Analyze Include Files - Kiem-Phong Vo (1992) (Correct) (3 citations)
lines that cpp processed from 23155 to 9076 -a **saving** of 61%The total **compilation** time of graphics.c
files included by F directly or indirectly. ffl **First**(F)include files directly included by F. ffl
which include files can be safely ignored for **compilation** and give a linear time algorithm to compute
www.research.att.com/~ciao/doc/incl.ps.gz

Whole-Program Optimization of Object-Oriented Languages - Dean (1996) (Correct) (18 citations)
code, sometimes resulting in an overall space **saving**s compared with a system that compiles all code
classes of objects that can appear at a call **site**. A selective specialization algorithm combines
ftp.cs.washington.edu/tr/1996/11/UW-CSE-96-11-05.PS.gz

Secrets of the Glasgow Haskell Compiler inliner - Jones, Marlow (1999) (Correct) (10 citations)
get large. Shao, for example, reports substantial **saving**s when using de Bruijn numbers (for types)
bindings of its free variables. But at the **inline site**, the compiler might know more about the dynamic
Considerable attention is paid to the effect on **register** allocation of larger basic blocks, which we do
www.research.microsoft.com/Users/simonpj/papers/inline.ps.gz

Lightweight Run-Time Code Generation - Leone, Lee (1994) (Correct) (34 citations)
this technique is monovariant because all call **sites** must supply a "compatible "division of the
to early or late stages of computation. ffl **Register** allocation assigns **registers** to program
Allocation Deferred **compilation** reduces **register pressure** but complicates **register** allocation. Fewer

www.cs.cmu.edu/afs/cs.cmu.edu/user/mleone/papers/lw-rtcg.ps

Documents 21 to 40 Previous 20 Next 20

Try your query at: Amazon Barnes & Noble Google (RI) Google (Web) CSB DBLP